Systems Engineering Technical Review (SETR): Navy's Acquisition and Development Process
and
Software Lessons Learned

Naval Air Systems Command
Systems Engineering Department
Software Engineering & Acquisition Management Division

Processes, Models, and Frameworks

- The following key processes, models, and frameworks are synergistic and when used together provide for efficiency and effectiveness of systems engineering and software acquisition and development:
  - **Systems Engineering Technical Review (SETR) Acquisition framework**
  - Basic Systems Engineering
  - IEEE/EIA 12207 *Software life cycle processes*
  - Capability Maturity Model Integration (CMMI®) Best Practices framework

Processes, Models, and Frameworks cont.

- While each of the four is normally taught separately, the establishment of a software engineering environment consistent with all four provides for an infrastructure that can be measured for effectiveness and improved to provide the warfighter with better quality products faster and cheaper – BEST VALUE
- They are based on best practices throughout industry, government, and academia and are designed to work with each other

Basic Systems Engineering

Systems Engineering

- A system is an integrated composite of people, products, and processes that provide a capability to satisfy a stated need or objective
- Systems Engineering (SE) is the effective application of scientific and engineering efforts to transform an operational need into a defined system configuration through the top-down iterative process of requirements definition, functional analysis and allocation, synthesis, optimization, design, test, and evaluation

Source: ***NAVAIR ACQUISITION GUIDE 2006/2007*, 20 Sep 2006.**

Systems Engineering Process

- Define customer needs and required functionality
- Plan the project
- Document the requirements
  - Validate and baseline requirements
- Develop the design based on the requirements
  - Validate that design meets all requirements and is baselined
- Produce the item based on the design
- Perform system validation
  - Ensure all requirements met and the required functionality performs as expected

Planning

Requirements Phase

Design Phase

Coding/Implementation

Testing

Based International Council on Systems Engineering (INCOSE) definition

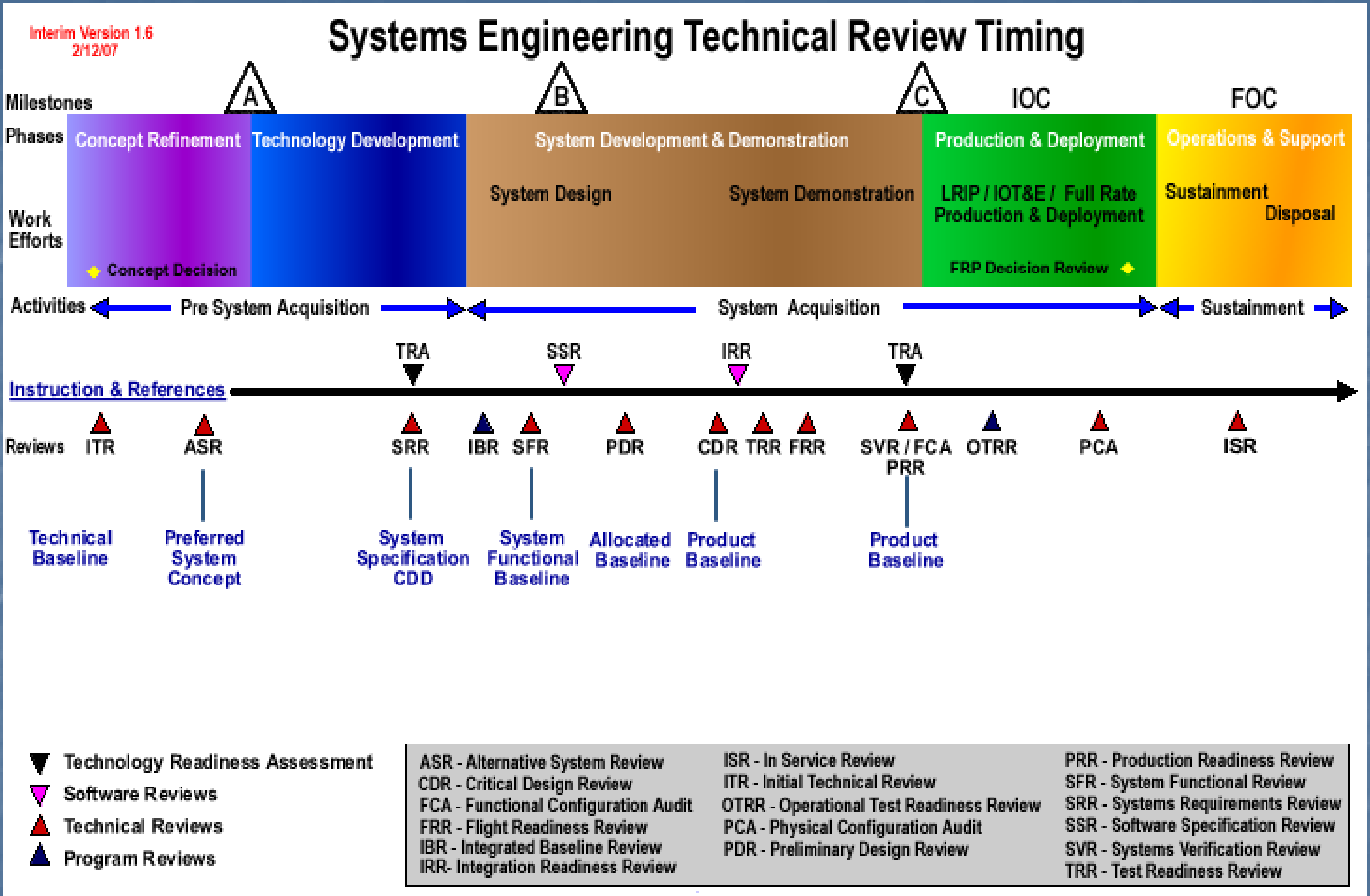
Systems Engineering Technical Review Process

SETR

- What is the SETR?
  - Systems Engineering Technical Review was developed by NAVAIR and has been adapted for use within the entire Navy
  - An iterative program timeline that maps the technical reviews to the acquisition process described in DOD 5000 documentation
    - Aligned with DODI 5000.02
  - Policy and process described in NAVAIRINST 4355.19D
    - Applies to all personnel supporting all NAVAIR and Aviation Program Executive Officer (PEO) programs involved with the design, development, test and evaluation, acquisition, in-service support, and disposal of naval aviation weapon systems and equipment
- Phases
  - Material Solution Analysis[1] – ends with Milestone A **(to begin technology development)**
  - Technology Development – ends with Milestone B **(to begin SD&D)**
  - Engineering & Manufacturing Development[2] – ends w/ Milestone C **(LRIP)**
  - Production & Deployment – ends with Initial Operating Capability **(IOC)**
  - Operations & Support (**maintenance**)

[1] Was previously called Concept Refinement

[2] Was previously called System Development & Demonstration



Interim Version 1.6
2/12/07

Systems Engineering Technical Review Timing

Milestones: A, B, C, IOC, FOC

Phases: Concept Refinement | Technology Development | System Development & Demonstration | Production & Deployment | Operations & Support

Work Efforts: ◆ Concept Decision | System Design | System Demonstration | LRIP / IOT&E / Full Rate Production & Deployment, FRP Decision Review ◆ | Sustainment, Disposal

Activities: ← Pre System Acquisition → | ← System Acquisition → | ← Sustainment →

Instruction & References

TRA ▼ | SSR ▼ | IRR ▼ | TRA ▼

Reviews: ▲ ITR | ▲ ASR | ▲ SRR | ▲ IBR | ▲ SFR | ▲ PDR | ▲ CDR | ▲ TRR | ▲ FRR | ▲ SVR / FCA PRR | ▲ OTRR | ▲ PCA | ▲ ISR

Technical Baseline | Preferred System Concept | System Specification CDD | System Functional Baseline | Allocated Baseline | Product Baseline | Product Baseline

▼ Technology Readiness Assessment
▼ Software Reviews
▲ Technical Reviews
▲ Program Reviews

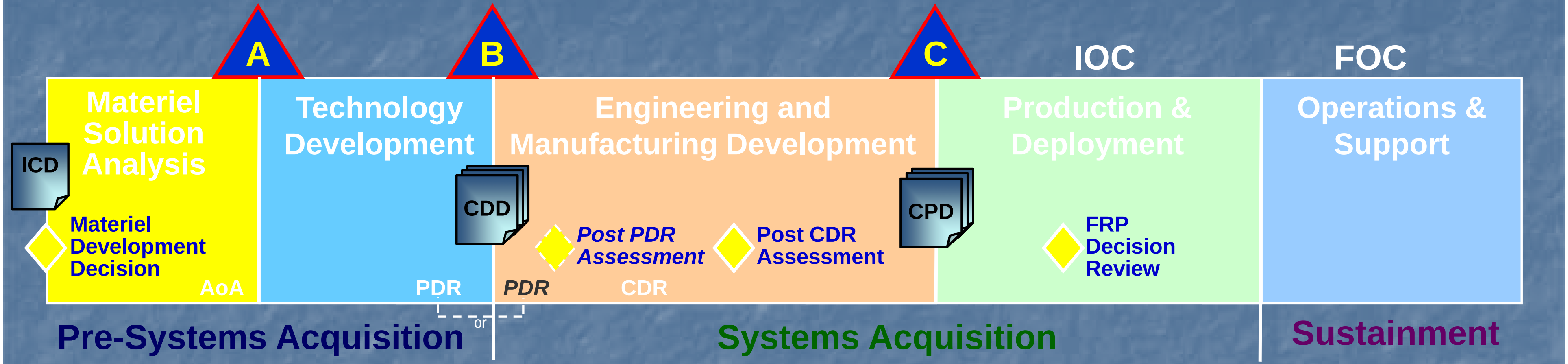
ASR - Alternative System Review
CDR - Critical Design Review
FCA - Functional Configuration Audit
FRR - Flight Readiness Review
IBR - Integrated Baseline Review
IRR- Integration Readiness Review

ISR - In Service Review
ITR - Initial Technical Review
OTRR - Operational Test Readiness Review
PCA - Physical Configuration Audit
PDR - Preliminary Design Review

PRR - Production Readiness Review
SFR - System Functional Review
SRR - Systems Requirements Review
SSR - Software Specification Review
SVR - Systems Verification Review
TRR - Test Readiness Review

Found at: **https://nserc.navy.mil/Pages/SETRTimeline.aspx**
Documented in: *NAVAIRINST 4355.19D*

Changes Based on new 5000.02



A B C IOC FOC

Materiel Solution Analysis

Technology Development

Engineering and Manufacturing Development

Production & Deployment

Operations & Support

ICD

Materiel Development Decision

AoA

CDD

PDR

or

PDR

Post PDR Assessment

Post CDR Assessment

CDR

CPD

FRP Decision Review

Pre-Systems Acquisition

Systems Acquisition

Sustainment

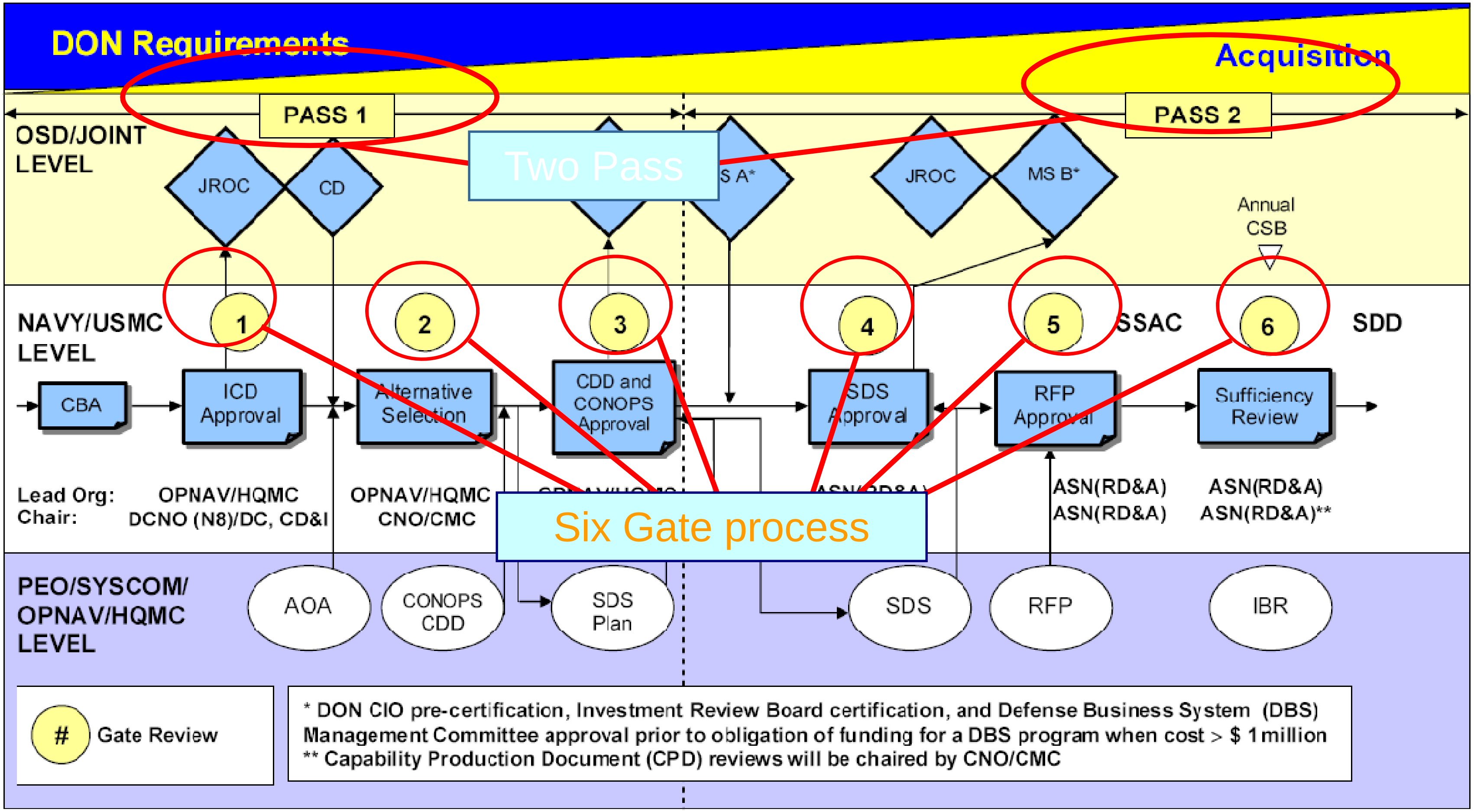
- Concept Refinement to Materiel Solution Analysis
- System Development & Demonstration back to Engineering & Manufacturing Development
- The Preliminary Design Review will be either before or after Milestone B as defined in the program’s acquisition strategy

Two Pass/Six Gate Process

Objective: Establish a disciplined and integrated process for requirements and acquisition decision-making within DON, endorsing or approving key Joint Capabilities Integration and Development Systems (JCIDS) and acquisition documents at Gate reviews, and facilitating decisions regarding required Navy and Marine Corps capabilities and acquisition of corresponding material solutions.

The SETR and the Two Pass/Six Gate process are complimentary

DON Requirements/Acquisition Two-Pass/Six-Gate Process with Development of a System Design Specification
(illustrated example for program initiation at Milestone A)



* DON CIO pre-certification, Investment Review Board certification, and Defense Business System (DBS) Management Committee approval prior to obligation of funding for a DBS program when cost > $ 1 million
** Capability Production Document (CPD) reviews will be chaired by CNO/CMC

AOA	Analysis of Alternative	IBR	Integrated Baseline Review
ASN(RD&A)	Asst Secretary of the Navy (Research, Development and Acquisition)	ICD	Initial Capabilities Document
CBA	Capabilities-Based Assessment	JROC	Joint Requirements Oversight Council
CD	Concept Decision	PEO	Program Executive Officer
CDD	Capability Development Document	RFP	Request for Proposal
CMC	Commandant of the Marine Corps	SDD	System Development & Demonstration
CNO	Chief of Naval Operations	SDS	System Design Specification
CONOPS	Concept of Operations	SSAC	Source Selection Advisory Council
CSB	Configuration Steering Board		
HQMC	Headquarters Marine Corps		

Major Phases of SETR

SETR Software Aspects

- Software is a key component in almost every SETR
- The NAVAIR Software Enterprise has developed a set of templates for each review
  - Defines the contents of the software section for the review
  - Includes a list of items to be covered along with the entrance criteria as it applies to software

Example of Items from SRR Template

- SW Development Team
- Integrated Mater Schedule Highlighted with Software Milestones
- Software Entrance Criteria
- Requirements Analysis and Allocation Methodology
- System Specifications Tree
- Contract Data Requirements List (CDRL)
- Software Development Strategy
- Software Development Process
- SW Safety, Information Assurance and Security requirements
- Software Supplier Management
- Software Measurement
- Software Risk Assessment with Mitigation Strategies
- Issues and Concerns

Systems Requirements Review (SRR)

SRR

- Technical assessment establishing the system specification of the system under review to ensure a reasonable expectation of being judged operationally effective & suitable
- Ensures the Capability Development Document (CDD), DOD Directives, statutory and regulatory guidance, and applicable public law has been correctly and completely represented in the system specification and can be developed within program cost and schedule constraints
  - Systems requirements are evaluated to determine whether they are fully defined and consistent with the mature system solution, and whether traceability of systems requirements to the CDD is maintained
- Assesses the performance requirements as captured in the system specification, and ensures the preferred system solution is:
  - Consistent with the system specification
  - Correctly captures derived and correlated requirements
  - Has well understood acceptance criteria and processes
  - Is achievable through available technologies resulting from the Technology Development phase

SRR-I and SRR-II

- For major programs going through both concept refinement/materiel solution analysis and technology development, there will be 2 SRRs
- SRR-I
  - Technical assessment establishing the specification of the system under review, previously represented by the Performance Based Specification (PBS), to continue the requirements decomposition process prior to MS B
- SRR-II
  - This review ensures the contractors participating in the Technology Development (TD) Phase understands that the requirements of the contract including the system specification, SOW, CDD, DoD Directives, statutory and regulatory guidance, and applicable public law has been correctly and completely represented in the system specification and can be developed within program cost and schedule constraints

SRR Software Products

- Software Development Plan (SDP)
  - Schedule
  - Processes
  - Software Development Environment
- System Specification
- Initial Software Requirements Description (SRD)
- Modeling and Simulation Plan
- Supporting Products
  - Systems Engineering Plan
  - Risk Management Plan
  - Measurement Plan
  - Quality Assurance Plan
- Measurement Data
  - Cost
  - Size
  - Requirements Volatility

Technical Readiness Assessment (TRA)

TRA

- A systematic metrics-based process that assesses the maturity of Critical Technology Elements (CTEs) by an independent panel of technical experts
- Applicable to all acquisition category (ACAT) programs per Secretary of the Navy Instruction 5000.2C
- May be combined with a SRR

Definitions

Technology Readiness Level Definitions

TRL Level	Criteria
1	**Basic principles observed and reported**: Transition from scientific research to applied research. Essential characteristics and behaviors of systems and architectures. Descriptive tools are mathematical formulations or algorithms.
2	**Technology concept and/or application formulated**: Applied research. Theory and scientific principles are focused on specific application area to define the concept. Characteristics of the application are described. Analytical tools are developed for simulation or analysis of the application.
3	**Analytical and experimental critical function and/or characteristic proof-of-concept**: Proof of concept validation. Active Research and Development (R&D) is initiated with analytical and laboratory studies. Demonstration of technical feasibility using breadboard or brassboard implementations that are exercised with representative data.
4	**Component/subsystem validation in laboratory environment**: Standalone prototyping implementation and test. Integration of technology elements. Experiments with full-scale problems or data sets.
5	**System/subsystem/component validation in relevant environment**: Thorough testing of prototyping in representative environment. Basic technology elements integrated with reasonably realistic supporting elements. Prototyping implementations conform to target environment and interfaces.

Definitions cont.

Technology Readiness Level Definitions continued

TRL Level	Criteria
6	**System/subsystem model or prototyping demonstration in a relevant end-to-end environment (ground or space**): Prototyping implementations on full-scale realistic problems. Partially integrated with existing systems. Limited documentation available. Engineering feasibility fully demonstrated in actual system application.
7	**System prototyping demonstration in an operational environment (ground or space**): System prototyping demonstration in operational environment. System is at or near scale of the operational system, with most functions available for demonstration and test. Well integrated with collateral and ancillary systems. Limited documentation available.
8	**Actual system completed and "mission qualified" through test and demonstration in an operational environment (ground or space)**: End of system development. Fully integrated with operational hardware and software systems. Most user documentation, training documentation, and maintenance documentation completed. All functionality tested in simulated and operational scenarios. Verification and Validation (V&V) completed.
9	**Actual system "mission proven" through successful mission operations (ground or space)**: Fully integrated with operational hardware/software systems. Actual system has been thoroughly demonstrated and tested in its operational environment. All documentation completed. Successful operational experience. Sustaining engineering support in place.

Integrated Baseline Review (IBR)

IBR

- Employed by Program Managers (PMs) throughout the life of projects where Earned Value Management (EVM) is required
- The IBR establishes a mutual understanding of the project Performance Management Baseline (PMB) and provides for an agreement on a plan of action to evaluate risks inherent in the PMB and the management processes that operate during project execution
- Assessment of risk within the PMB and the degree to which the following have been established:
  - Technical scope of work
  - Project schedule key milestones
  - Resources
  - Task
  - Rationale
  - Management Processes

System Functional Review (SFR)

SFR

- Technical assessment establishing the system functional baseline of the system under review to ensure a reasonable expectation of being judged operationally effective and suitable
- Assesses the decomposition of the system specification to system functional specifications derived from use case analysis
- Functional requirements for operations and maintenance are assigned to sub-systems, hardware, software, or support after detailed reviews of the architecture in the environment it will be employed
    - The system’s lower level performance requirements are evaluated to determine whether they are fully defined and consistent with the mature system concept, and whether traceability of lower-level systems requirements to top-level system performance and the CDD is maintained
- The development of representative operational use cases for the system
- The SFR determines whether the systems functional definition is fully decomposed to its lower level, and that the Team is prepared to start preliminary design for hardware
- Risk Management, Measurement, Quality Assurance, & Configuration Management processes fully functional

SFR Products

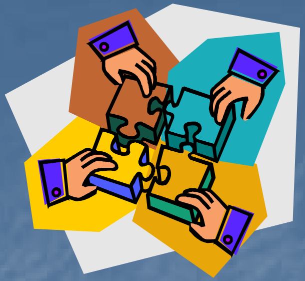
- Updates to SRR products
  - SDP
  - System Specification
  - Modeling and Simulation Plan
  - Supporting Products
    - Systems Engineering Plan
    - Risk Management Plan
    - Measurement Plan
    - Quality Assurance Plan
    - Cost and Size Estimates
- Updated (more detail) Software Requirements Description (SRD)
- Interface Control Documents
- Draft Test Plan
- Measurement Data
  - Cost
  - Size
  - Requirements Volatility

Software Specification Review (SSR)

SSR

- Technical assessment establishing the software requirements baseline to ensure the preliminary design and ultimately the software solution has a reasonable expectation of being judged operationally effective and suitable
  - The software's lower level performance requirements are fully defined and consistent with a mature system concept, and traceability of lower-level software requirements to top-level system performance and the CDD is maintained
- A review of the finalized Computer Software Configuration Item (CSCI) requirements and operational concept
- Software Requirements Specification (SwRS) or Software Requirements Description (SRD); Interface Requirements Specification(s) (IRS) or Software Interface Requirements Description (SIRD); Software Integration Plan; and the user's Concept of Operation Description or User Documentation Description form a satisfactory basis for proceeding into preliminary software design

SSR Products



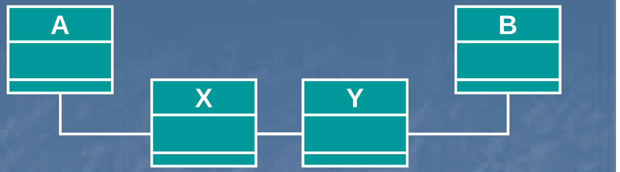
- Updates to SRR and SFR products
- Final Software Requirements Specification (SwRS) or Software Requirements Description (SRD)
- Requirements Verification Matrix
  - CDD to specifications to SwRS or SRD
- Interface Control Documents (ICDs) and Interface Requirements Specification (IRS) or Software Interface Requirements description (SIRD)
- Declassification, Anti-Tamper, Open Architecture, and Information Assurance requirements
- Completed Test Plan
- Software Integration Plan
- Measurement Data
  - Cost
  - Size
  - Requirements Volatility

Preliminary Design Review (PDR)

PDR

- Technical assessment establishing the physically allocated baseline to ensure that the system has a reasonable expectation of being judged operationally effective and suitable
- Assesses the allocated design captured in subsystem product specifications for each configuration item in the system and ensures that each function, in the functional baseline, has been allocated to one or more system configuration items
- Subsystem specifications for hardware and software, along with associated Interface Control Documents (ICDs), enable detailed design or procurement of subsystems
- A successful review is predicated on the Team’s determination that the subsystem requirements, subsystem preliminary design, results of peer reviews, and plans for development and testing form a satisfactory basis for proceeding into detailed design and test procedure development

PDR Products



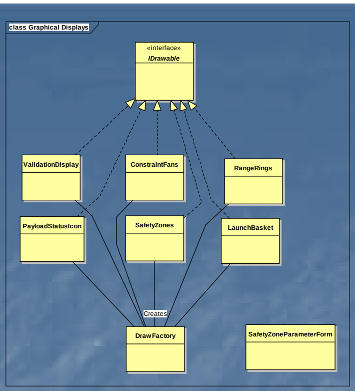
- Updates to SRR, SFR, and SSR products
- Top Level Software Design Description and/or Software Architecture Description
- Completed Test Plan
- Draft Test Procedures
- Traceability from design documentation to subsystem test requirements
- Representative mission profiles
- Measurement Data
  - Size
  - Defects
  - Requirements Volatility

Critical Design Review (CDR)

CDR

- Technical assessment establishing the build baseline to ensure that the system has a reasonable expectation of being judged operationally effective and suitable
- Assesses the final design as captured in product specifications for each configuration item in the system, and ensures that item has been captured in detailed documentation
- Product specifications for software enable coding of the Computer Software Configuration Item (CSCI)
- A successful review is predicated on the Team's determination that the subsystem requirements, subsystem detail design, results of peer reviews, and plans for testing form a satisfactory basis for proceeding into implementation, demonstration, and test

CDR Products

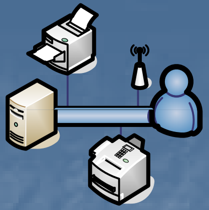


- Updates to SRR, SFR, SSR, & PDR products
  - All specifications and requirements documentation are complete/stable
- Detailed Software Design Description
- Units Test Procedures
- Traceability Verification Matrix
  - CDD to specifications to requirements documentation to design to test procedures
  - Traceability in both directions
- Measurement Data
  - Size
  - Defects
  - Maturity
  - Requirements Volatility

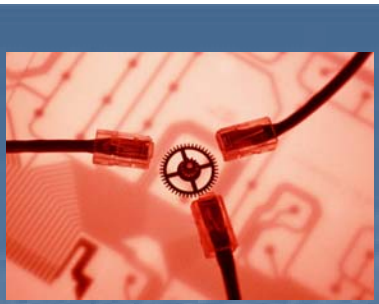
Integration Readiness Review (IRR)

IRR

- Technical assessment establishing the configuration to be used in integration test to ensure that the system has a reasonable expectation of being judged operationally effective and suitable
- A product and process assessment to ensure that hardware and software or software components are ready to begin integrated configuration item (CI) testing
  - Assess prior component or unit level testing adequacy, test planning, test objectives, test methods and procedures, scope of tests, and determines if required test resources have been properly identified and coordinated to support planned tests
  - Verifies the traceability of planned tests to program, engineering data, analysis, and certification requirements
- Testing is based upon the Test Plan (TP)
  - Begun in requirements phase and completed during design
- Conducted after the test and/or validation procedures are complete and unit level testing is complete



IRR Products



- Updates to SRR, SFR, SSR, PDR, & CDR products
- Approved Integration Test Plan
- Approved Integration Test Procedures
- Format for Integration Test Report
- Completed Integration Test Verification Matrix
- Measurement Data
  - Quality/maturity
  - Defects

Test Readiness Review (TRR)

TRR

- Technical assessment establishing the configuration used in test to ensure that the system has a reasonable expectation of being judged operationally effective and suitable
- TRR is a multi-disciplined product and process assessment to ensure that the subsystem, system, or systems of systems has stabilized in configuration and is ready to proceed into formal test
  - Assesses prior unit level and system integration testing adequacy, test planning, test objectives, test methods and procedures, scope of tests, and determines if required test resources have been properly identified and coordinated to support planned tests
  - Verifies the traceability of planned tests to program, engineering, analysis, and certification requirements
  - Determines the completeness of test procedures and their compliance with test plans and descriptions
  - Assesses the impact of known discrepancies to determine if testing is appropriate prior to implementation of the corrective fix
- The TRR process is equally applicable to all tests in all phases of an acquisition program

TRR Products

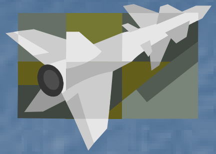
- Updates to SRR, SFR, SSR, PDR, CDR, and IRR products
- Traceability Analysis
- Approved Test Plan
- Approved Test Procedures
- Format for Test Report
- Completed Test Verification Matrix
- Software Version Document
- Measurement Data
  - Quality/maturity
  - Defects

Flight Readiness Review (FRR)

FRR

- Technical assessment establishing the configuration used in flight test to ensure that the system has a reasonable expectation of being judged operationally effective and suitable
- Assesses the system and test environment to ensure that the system under review can proceed into flight test with:
  - NAVAIR airworthiness standards met
  - Objectives clearly stated
  - Flight test data requirements clearly identified
  - Acceptable risk management plan defined and approved
- Ensures that:
  - Proper coordination has occurred between engineering and flight test
  - All applicable disciplines understand and concur with:
    - The scope of effort that has been identified
    - How this effort will be executed to derive the data necessary (to satisfy airworthiness and test and evaluation requirements) to ensure the weapon system evaluated is ready to proceed to flight test

FRR Products



- Updates to SRR, SFR, SSR, PDR, CDR, IRR, & TRR products
- Approved Flight Test Plan
- Approved Test Points/Flight Scenarios/Knee Cards
- Format for Flight Test Report
- Software Version Document
- Measurement Data
  - Quality/maturity
  - Defects/Test Hour

Operational Test Readiness Review (OTRR)

OTRR

- Multi-disciplined product and process assessment to ensure that the system under review can proceed into Operational Test and Evaluation (OT&E) with a high probability the system will successfully complete operational testing
- Successful performance during Operational Evaluation (OPEVAL) generally indicates the system being tested is effective and suitable for Fleet introduction
    - The decision to enter production may be based on this successful determination
- Of critical importance to this review is the understanding of available system performance to meet the Capability Production Document (CPD)
- Operational requirements defined in the CPD must match the Requirements tested to in the Test and Evaluation Master Plan (TEMP)

OTRR Products

- Updates to SRR, SFR, SSR, PDR, CDR, IRR, TRR, & FRR products
- Approved Test & Evaluation Master Plan (TEMP)
- Approved Test Points/Flight Scenarios
- Format for Operational Test Report & Quick-look Report
- Measurement Data
  - Quality/maturity
  - Defects/Test Hour

NAVAIR-Specific Lessons Learned

Requirements Lessons Learned

- Some developers tend to resist documenting requirements in a requirements traceability tool
  - Inability to trace requirements back to customer's/sponsor's requirements
  - Requirements creep – adding requirements not needed to meet user's/customer's desires
- Lack of concurrence among the **stakeholders** of the requirements (collaboration)
  - Key contributor to requirements instability, which leads to cost and schedule problems
- Requirements too loose/broadly written, making the requirements decomposition more difficult

Requirements Lessons Learned cont.

- Tendency to begin preliminary design before requirements done or maturity of the requirements verified
  - Can result in a lot of rework
  - If the requirements are not fully defined, then your estimates for both cost and schedule will be inaccurate
- Resistance to having a requirements change control board early in the requirements phase
- Lack of requirements stability measure (metric)
- Requirements phase too little time

Design Lessons Learned

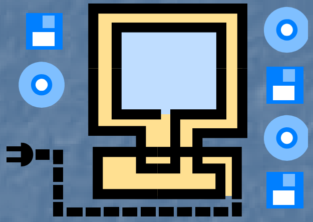
- Tendency to combine preliminary design and detailed design into a single phase to save time
  - Peer reviews tend to have more defects because designs not thought out well
  - Tendency to begin coding/production before design maturity verified
- Those who use a design tool, tend to do better designs, especially if requirements were managed with a tool
- See some confusion between architecture definition and design
  - These developers tend to begin coding early and call it prototyping

Agile Programming

WE'RE GOING TO TRY SOMETHING CALLED AGILE PROGRAMMING.
THAT MEANS NO MORE PLANNING AND NO MORE DOCUMENTATION. JUST START WRITING CODE AND COMPLAINING.
I'M GLAD IT HAS A NAME.
THAT WAS YOUR TRAINING.

Coding Lessons Learned

- Size tends to go up and amount of reuse tends to go down as a function of time
  - **Add growth into planning**
- Tend to underestimate the amount of effort required for reusing code
  - Consider if new code will be cheaper
  - Can be as much as 1.5 times the cost of new code



- Auto code generator use increasing
  - Problems/defects come from humans
  - **Most effective when used in conjunction with a design tool**
- Peer reviews are extremely important during this phase
  - Detects problems when they are cheaper to fix
- Integrated unit testing tends to get shortened when schedules slide
  - Consider sliding schedule or reducing content if schedule cannot slide
- Resource planning (labs, tools, and people) not necessarily well thought out, especially when there is a hardware-in-the-loop lab

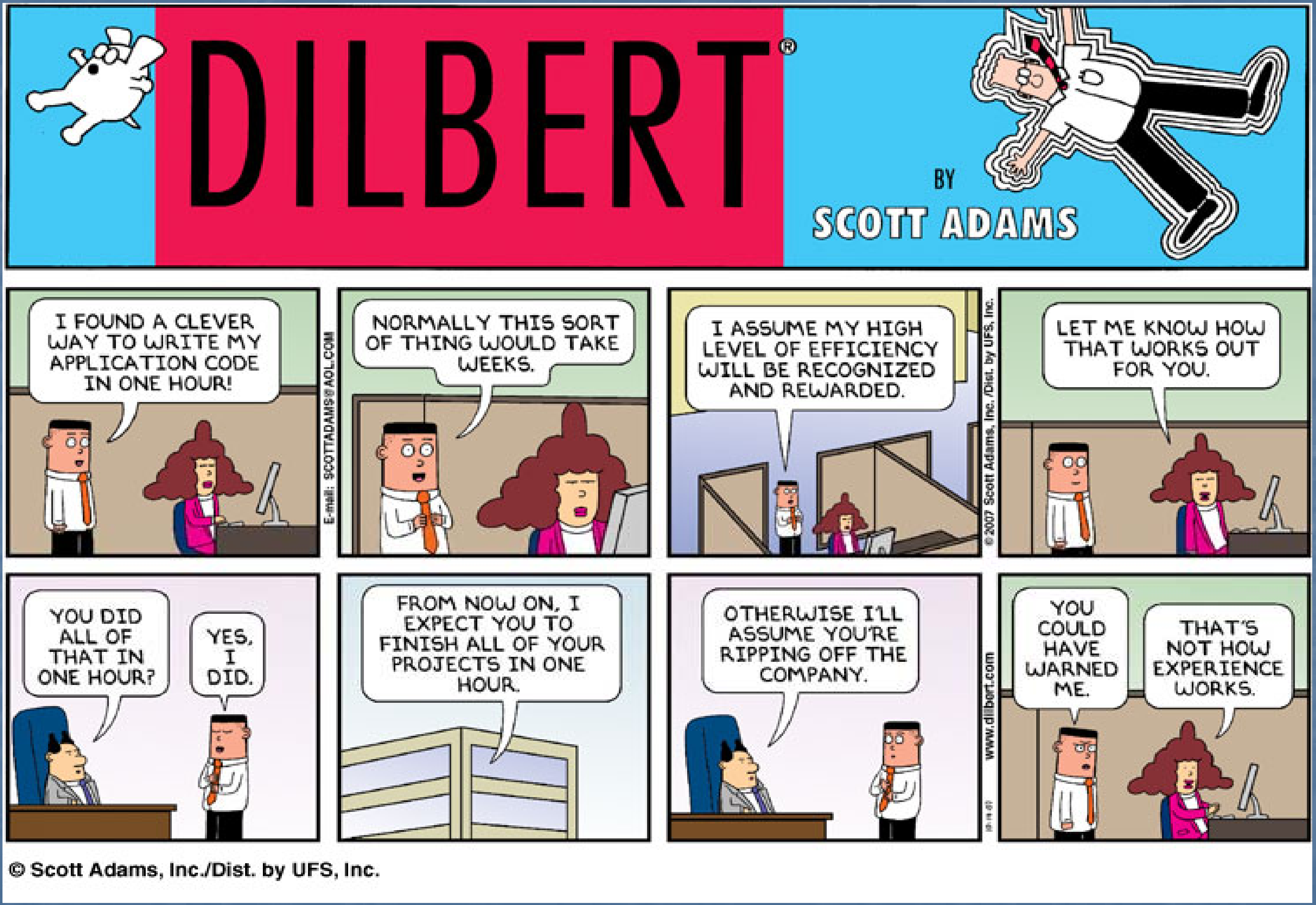
Testing Lessons Learned

- Late development of test plans and procedures
  - May not be able to get all the resources needed
  - May not have the proper review or secured commitment to the testing
- Too little time for testing
  - Have to rush through tests and may overlook problems
  - Have to cut out some of the tests
  - Have to test concurrent with development, which may cause retest when tested areas of the software are changed
- Automated testing promotes
  - Better coverage during testing
  - More efficient use of staff
  - Repeatability of testing
  - Efficient usage of test facilities during 2nd & 3rd shifts

Testing Lessons Learned cont.

- Don’t document test sessions
  - Anomalies discovered in the test session may get overlooked
  - Don’t have data to show what has been tested to support decisions
    - Readiness for next step
    - Flight clearances
    - Production decisions
  - Can’t take credit for the testing performed and may have to redo testing
- Wrong configuration tested
- Contention for test facilities due to poor up-front planning
  - Schedule delays
  - Removal of tests

Schedule Compliance


DILBERT®
BY SCOTT ADAMS
I FOUND A CLEVER WAY TO WRITE MY APPLICATION CODE IN ONE HOUR!
E-mail: SCOTTADAMS@AOL.COM
NORMALLY THIS SORT OF THING WOULD TAKE WEEKS.
I ASSUME MY HIGH LEVEL OF EFFICIENCY WILL BE RECOGNIZED AND REWARDED.
© 2007 Scott Adams, Inc./Dist. by UFS, Inc.
LET ME KNOW HOW THAT WORKS OUT FOR YOU.
YOU DID ALL OF THAT IN ONE HOUR?
YES, I DID.
FROM NOW ON, I EXPECT YOU TO FINISH ALL OF YOUR PROJECTS IN ONE HOUR.
OTHERWISE I'LL ASSUME YOU'RE RIPPING OFF THE COMPANY.
www.dilbert.com
YOU COULD HAVE WARNED ME.
THAT'S NOT HOW EXPERIENCE WORKS.
© Scott Adams, Inc./Dist. by UFS, Inc.

Dilbert is copywrited by Scott Adams, Incorporated and is provided for non-commercial use by United Features Syndicate, Incorporated.

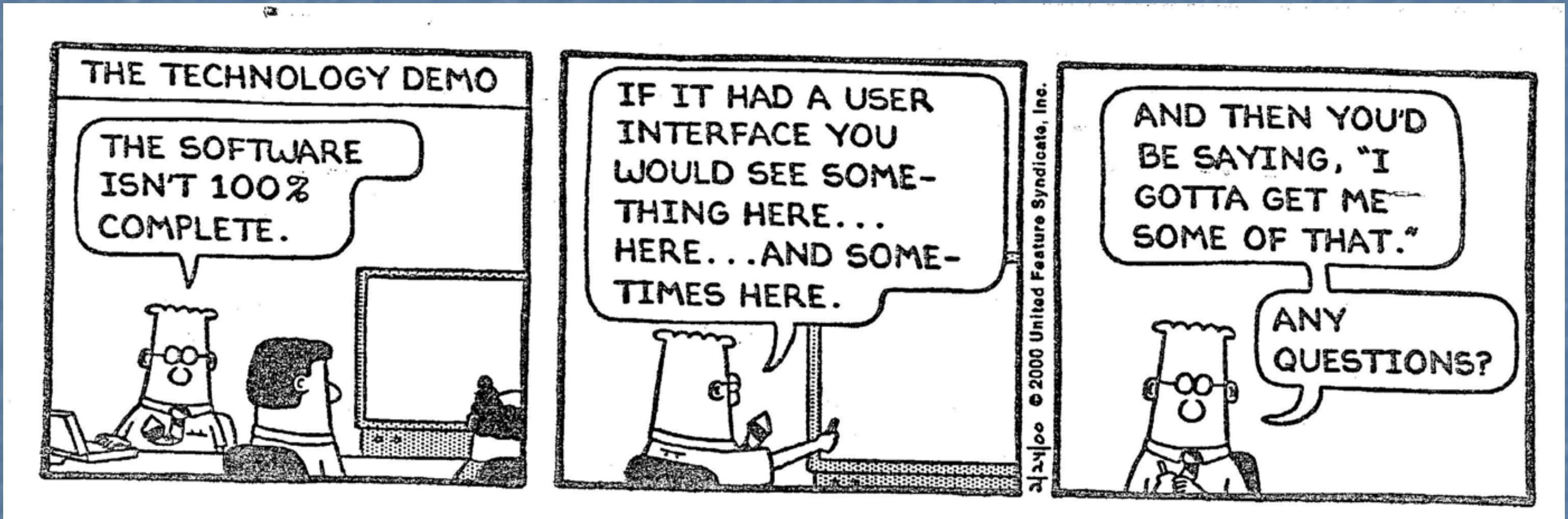
Schedule Issues

- Beginning tasks before maturity or readiness has been verified rarely if ever saves time and funding
  - JSF[1]
    - The JSF started **system development before requisite technologies were ready**, **started manufacturing test aircraft before designs were stable**, and **moved to production before flight tests have adequately demonstrated that the aircraft design meets performance and operational suitability requirements**.
  - FCS[2]
    - The current FCS practice is to **overlap builds more than the traditional spiral model does** with software developers reporting that **evolving requirements have caused them to interpret and implement changes in requirements well into the design and code phases**, compromising the amount of time allotted for testing

1 Source: GAO-08-388, *JOINT STRIKE FIGHTER: Recent Decisions by DOD Add to Program Risks*. March 2008

2 Source: GAO-08-409, *DEFENSE ACQUISITIONS : Significant Challenges Ahead in Developing and Demonstrating Future Combat System's Network and Software*. March 2008

Schedule Driven Demos


THE TECHNOLOGY DEMO
THE SOFTWARE ISN'T 100% COMPLETE.
IF IT HAD A USER INTERFACE YOU WOULD SEE SOMETHING HERE... HERE...AND SOMETIMES HERE.
© 2000 United Feature Syndicate, Inc.
3/24/00
AND THEN YOU'D BE SAYING, "I GOTTA GET ME SOME OF THAT."
ANY QUESTIONS?

Dilbert is copywrited by Scott Adams, Incorporated and is provided for non-commercial use by United Features Syndicate, Incorporated.

Take your time and make sure the product is mature before going to the next step

Schedule Lessons Learned

- **Tendency for schedules to be too aggressive for the amount of work to be performed**
- **“When faced with unrealistic schedules, engineering teams often behave irrationally”***
  - Race through requirements
  - Produce a superficial design
  - Rush into coding
  - Software tends to be delivered later than if it were developed with a rational plan
- Lack of staffing allocation across the schedule
  - Most common problem is staff brought in too late
- Overlapping of phases (to save time) and beginning next phase with an immature product
- Critical path is not evident
  - Impacts risk management
  - Puts you in reaction mode (chaotic) because you did nothing to mitigate
- Insufficient detail to assess risks
- Unknown linkages and interdependencies of the tasks identified

* Source: Humphrey, Watts S. “Winning in Software: An Executive Strategy.” Addison-Wesley, 2002.

Software Suppliers Lessons

- In an article in CrossTalk Magazine* on software acquisition in the Army, Edgar Dalrymple, the Program Manager for the Future Combat Systems Brigade Combat Team and Associate Director of Software and Distributed Systems, when answering a question making one change in the way the government procures software:

"The government, at least the Army, needs to stop buying software exclusively from the traditional defense contracting base. These companies have the overhead costs of manufacturing companies, yet software development should carry a far smaller overhead burden. Most defense contractors are still managed by manufacturing engineers or business managers."

* Source: Starrett, Elizabeth. "Software Acquisition in the Army." CrossTalk May 2007.

Subcontractor Management Lessons

- In major contracts where the prime has subcontracted out subsystems, we have seen:
  - Software requirements regarding process maturity and development of deliverables not passed down to the subcontractor
  - Prime subcontractor management personnel not experienced in software development
    - Results in latency of requirements changes to software
      - Rework
      - Schedule delay
    - Requirements volatility

Summary

- The Systems Engineering Technical Review process is an iterative program timeline that maps to DODI 5000.02 acquiaition timeline and is compatible with:
  - Basic Systems Engineering
  - CMMI
  - IEEE 12207
- There are many software development lessons that NAVAIR has learned from its participation in SETR events